Bruce Xi

(236)-777-8218 • brucexi99@outlook.com • bruce-shidi-xi
brucexi999

Education

The University of British Columbia **Vancouver, BC**
Master of Engineering in Electrical and Computer Engineering, GPA 4.0 *2021–2024*

- Relevant courses: Deep Learning, ML Hardware Accelerator, Computer Architecture, Digital System Design, Embedded System, VLSI, IC Testing and Reliability

Imperial College London **London, UK**
Bachelor of Engineering in Materials Science and Engineering *2018–2021*

- Graduated with First-Class Honours
- Obtained Dean's List for three consecutive years (2018-2021)

Experience

Motorola Solutions **Vancouver, BC**
Design Validation Co-op *May-Dec. 2022*

- Conducted extensive camera tests in various settings, ensuring accuracy and precision
- Engineered Python-based software for test automation and data analysis, enhancing test efficiency with up to 90% automation
- Utilized FFmpeg for video processing
- Collaborated within a team using Git, Confluence, and Jira to optimize workflow efficiency

Project

Multi-agent Deep Reinforcement Learning for VLSI Routing **May-Oct. 2023**

- Self-taught DRL and VLSI global routing, demonstrating initiative and commitment
- Developed a multi-agent DRL router using PyTorch and RLlib, the router was equipped with a custom GNN for enhanced policy generalization
- The proposed router outperformed an A* baseline by 2.6%

Silicon Physical Design **Sept.-Dec. 2022**

- Designed the physical layout of a 3-input NAND gate using Cadence Virtuoso, the layout was DRC error free and optimized for timing and area
- Synthesized an RTL design using Cadence Encounter RTL Compiler
- Placed and routed the post-synthesis netlist using Cadence Innovus
- Simulated the circuit after placement and routing using Cadence Virtuoso

Embedded System Development **Jan.-Apr. 2023**

- Engineered key components of an embedded system including a 4-way set-associative cache controller and a DRAM controller using Verilog. The cache reduced the runtime of a benchmark by 43%
- Implemented the system on an FPGA with a provided soft microcontroller
- Developed software and firmware in C that interacted with hardware using SPI, IIC, and CAN protocol
- Utilized hardware timer interrupt and designed a snake game software that ran on the embedded system

CPU Architecture Design **June-Sept. 2022**

- Architected a 16-bit RISC CPU from the ground up using Verilog, integrating pivotal components such as FSM, datapath, RAM, and I/O interfaces
- The CPU supported 13 diverse instructions encompassing ALU operations, memory access, and branching mechanisms
- Successfully deployed the system onto an FPGA and validated the design's capabilities by executing a test program

Skills

Hardware: Verilog, FPGA, ModelSim, Quartus, Cadence

Software: Python, C, Assembly, Linux

DevOps Tools: Confluence, Jira, Git, GitHub